mesh_evaluation

The estimation of the spacings of the mesh along y direction is performed through the Python function `mesh_evaluation.py` inside the folder `1-pre_processing/`. *mesh_evaluation* can help in the pre-processing of Temporal Turbulent Boundary Layer (TTBL) simulations as well as Channel flows. The scaling employed by Incompact3d for channel flows is based on the laminar Poiseuille flow. For more details on the reference scales for channels, see the *Quick Incompact3d guide*. The subroutine `stretching_mesh_y` (that can be found inside the file `mesh_subs.py`) derives directly from the default subroutine of Incompact3d, `stretching`. Mesh evaluation is driven by the file `input.i3d` used to run simulations.

Generated files

`mesh_evaluation.py` prints to screen useful information for the simulations, as well as saving them to `.txt` files. We list the main ones as following:

In `sim_settings.txt` we store:

- non-dimensional domain dimensions, L_x^+, L_y^+, L_z^+;
- numerical parameters (CFL, Péclet, Numerical Fourier and stability parameter, Co, $\mathcal{D}$, Pe, S). For numerical stability, it is recommended to maintain $Co < 1$, $D < 0.5$, $Pe < 2$ and $S < 1$;
- mesh spacings Δx^+, Δy_w^+, Δy_δ^+, Δz^+ (w: first element at the wall, δ: element at the interface (TTBL) or at the centerline (Channel));
- number of mesh nodes `npvis` in the viscous sublayer at c_f peak (TTBL) or at steady state (Channel);
- shear velocity u_τ;
- number of mesh nodes `npsl` in the initial shear layer (TTBL);
- approximate initial shear layer momentum thickness θ_{sl} (TTBL);
- initial shear layer thickness δ_{99}^+ (TTBL);
- total number of mesh points, n_{tot};
- number of snapshots for a single flow realization, n_{snap};
- total memory requirement (storage), `mem_tot`;
- estimated CPUh to complete the simulation.

In the file `mesh_y.txt` instead:

- dimensional mesh spacings in y direction: Δy.
- Growth Rates (GRs) along y of grid elements (ratio of heights of adjacent elements).
- Aspect Ratios (ARs) in the xy plane of grid elements (ratio width/height of a single element).

A note on TTBL setup

It is worth noticing that in a TTBL simulation, the peak c_f value constraints the height of the first cell at the wall Δy_w, as in standard CFD simulations. However, the decrease of c_f along the simulation (and thus the increase in viscous length δ_ν) imposes a constraint for the domain dimensions L_x, L_y, L_z since they appear progressively "smaller" (their non-dimensional counterparts decreases). Too low values of L_x^+,L_y^+,L_z^+ must be avoided in order to do not enforce a too strong periodicity to the turbulent structures.

Pre-processing quantities

Some of the calculated quantities in `mesh_evaluation.py` for TTBL and Channel.
The reference velocity U_{ref} is U_w for a TTBL while it is U_B for a Channel.

Estimated Courant-Friedrichs-Lewy number (< 1)

$$Co = \frac{U_{ref}\Delta t}{\Delta x}$$

Estimated numerical Fourier number (< 0.5)

$$\mathcal{D} = \frac{\nu \Delta t}{\Delta x^2}$$

Estimated numerical Péclet number (< 2)

$$Pe = \frac{U_{ref}\Delta x}{\nu}$$

Estimated stability parameter (Thompson et al. (1985)) (< 1)

$$S = \frac{{U_{ref}}^2 \Delta t}{2\nu}$$

Pre-processing quantities for TTBL

Some of the calculated quantities in `mesh_evaluation.py` for a TTBL.

Shear velocity at IC

$$\frac{\partial U}{\partial y} = -\frac{U_w}{4\theta_{sl}} \cdot \frac{1}{\cosh^2\left(\frac{D}{2\theta_{sl}}\right)}$$

$$u_\tau = \sqrt{\nu \left| \frac{\partial U}{\partial y} \right|}$$

Shear velocity at peak friction coefficient

$$u_\tau = U_w \sqrt{\frac{c_f}{2}}$$

Initial velocity profile

$$U_0^+(y) = \frac{U_w^+}{2} + \frac{U_w^+}{2} \tanh\left[\frac{D}{2\theta_{sl}}\left(1 - \frac{y}{D}\right)\right]$$

Initial shear layer momentum thickness

$$\theta_{sl} \approx \frac{54\nu}{U_w}$$